Oracle® Banking Corporate Accounts Cloud Service

EOD Configuration User Guide

Release 14.8.1.0.0
G44785-01
October 2025

Oracle Banking Corporate Accounts Cloud Service EOD Configuration User Guide, Release 14.8.1.0.0

G44785-01

Contents

Preface

1 EOD Configuration

2 Oracle Banking Corporate Accounts Batch Jobs and APIs

3 Batch Description

A Customize Statements and Advices

Index

Preface

Purpose

The **EOD Configuration User Guide** helps you configure the batch processes to be executed at the **End of Day (EOD)**. You can find the list and descriptions of the Batch Jobs, API End points, and Functional Activity (FA) codes required to perform the EOD processing.

Audience

This user guide is intended for users who carry out the following roles within a bank.

Table 1 User Roles

User Role	Function
Back office clerk	Input functions for contracts
Back office managers/officers	Authorization functions
End of Day operators	Process at the End of Day or the Beginning of Day
Financial Controller/Product Managers	Generation of reports
Product Managers	Product definition and authorization

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Related Documents

The related documents are as follows:

- *Corporate Accounts User Guide*
- *Account Configurations User Guide*
- *Oracle Banking Common Core User Guide*

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Screenshot Disclaimer

Personal information used in the interface or documents is dummy and does not exist in the real world. It is only for reference purposes.

Acronyms and Abbreviations

Table 2 Commonly Used Acronyms and Abbreviations

Abbreviation	Definition
API	Application Programming Interface
EOD	End of Day
BOD	Beginning of Day
MCUT	Mark Cut-Off
EOTI	End of Transaction Input
EOFI	End of Financial Input

Basic Actions

This basic actions that can be performed on a screen are described in the following table.

Table 3   Basic Actions

Action	Description
Approve	Approve the initiated record. This option displays when the user clicks **Authorize**.
Audit	View the maker details, checker details, and record status.
Authorize	Authorize the record created. A maker of the screen is not allowed to authorize the record. Only a checker can authorize a record, created by a maker.
Cancel	Cancel the performed action.
Close	Close a record. This action is available only when a record is created.
Collapse All	Hide the details in the sections. This option displays when the user clicks **Compare**.
Compare	View the comparison through the field values of old record and the current record. This option displays in a widget when the user clicks **Authorize**.
Confirm	Confirm the performed action.
Expand All	Expand and view all the details in a section. This option displays when the user clicks **Compare**.
New	Add a new record. When the user clicks **New**, the system displays a new record to specify the required data.
OK	Confirm the details on the screen.
Reject	Reject the record created. A maker of the screen is not allowed to reject the record. Only a checker can reject a record, created by a maker.
Save	Save the details entered or selected in the screen.
Unlock	Update the details of an existing record. System displays an existing record in the editable mode.
View	View the record details in a particular modification stage. This option displays in the widget when the user clicks **Authorize**. This option is also displayed in the Tile menu.
View Difference only	View a comparison through the field element values of old record and the current record, which has undergone changes. This option is displayed when the user clicks **Compare**.

Note

The user must specify values for all the mandatory fields and they are marked as **Required** in the User Interface.

1

EOD Configuration

This topic provide information about the EOD Configuration processes.

The end-of-day (EOD) batch process runs at the end of each business day. It processes all the transactions that occur during the day. The process records and reconciles all transactions accurately. It helps maintain the integrity of the bank's financial data and ensures customers accounts are up-to-date.

The EOD batch process performs the following:

- Consolidate all the transactions
- Verify the accuracy of the data
- Update the records in the respective General Ledger (GL)

This topic contains the following subtopics:

- Mapping Functional Activity Code
  The topic provides information to map functional activity code to user roles who perform different EOD operations.
- Upload and Create EOD Work Flow Definition
  This topic describes the systematic instructions to upload DSL in **Business Process Maintenance**.
- Configure EOD
  This topic describes the systematic instructions to configure EOD (End of Day) operations.
- Invoke EOD
  This topic describes the systematic instructions to run the EOD for a branch.

1.1 Mapping Functional Activity Code

The topic provides information to map functional activity code to user roles who perform different EOD operations.

The user role performing these EOD operations should maintain the corresponding functional activity code listed in the table.

Functional Activity Code	**EOD Operations**
CMC_FA_SUBMENU_1_WORKFLOW_MAINT	Upload and Create EOD Workflow definition
CMC_UA_CORE_BRANCH_EOD_MAP	Configure EOD Workflow
CMC_FA_BRANCH_EOD_PROCESS	Invoke EOD Workflow
CMC_FA_CORE_BRANCH_EOD_RETRY	View End of Cycle Processes

> **Note**
>
> To know how to map a functional activity code to a user role, see the **Oracle Banking Security Management System User Guide**.

1.2 Upload and Create EOD Work Flow Definition

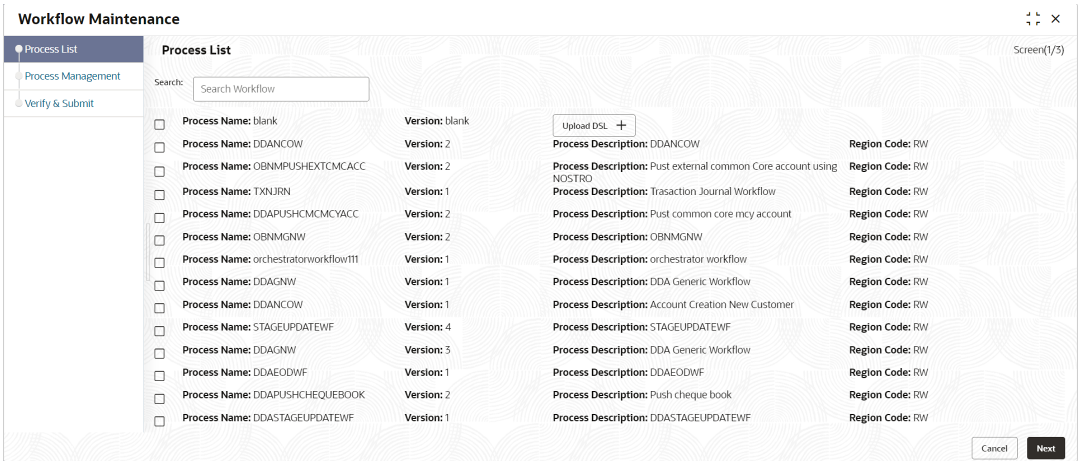
This topic describes the systematic instructions to upload DSL in **Business Process Maintenance**.

The `DDAEODWF.json` file is a standard work-flow process definition script for Oracle Banking Accounts. It lists the batch tasks that automatically execute in a sequence.

1. Click **Tasks**, and under **Tasks**, click **Business Process Maintenance** to import, create or modify batch process definitions.

   The **Workflow Maintenance** page displays the **Process List** screen.

   Figure 1-1 Workflow Maintenance - Process List

☐	**Process Name:** blank	**Version:** blank		
☐	**Process Name:** DDANCOW	**Version:** 2	**Process Description:** DDANCOW	**Region Code:** RW
☐	**Process Name:** OBNMPUSHEXTCMCACC	**Version:** 2	**Process Description:** Pust external common Core account using NOSTRO	**Region Code:** RW
☐	**Process Name:** TXNJRN	**Version:** 1	**Process Description:** Trasaction Journal Workflow	**Region Code:** RW
☐	**Process Name:** DDAPUSHCMCMCYACC	**Version:** 2	**Process Description:** Pust common core mcy account	**Region Code:** RW
☐	**Process Name:** OBNMGNW	**Version:** 2	**Process Description:** OBNMGNW	**Region Code:** RW
☐	**Process Name:** orchestratorworkflow111	**Version:** 1	**Process Description:** orchestrator workflow	**Region Code:** RW
☐	**Process Name:** DDAGNW	**Version:** 1	**Process Description:** DDA Generic Workflow	**Region Code:** RW
☐	**Process Name:** DDANCOW	**Version:** 1	**Process Description:** Account Creation New Customer	**Region Code:** RW
☐	**Process Name:** STAGEUPDATEWF	**Version:** 4	**Process Description:** STAGEUPDATEWF	**Region Code:** RW
☐	**Process Name:** DDAGNW	**Version:** 3	**Process Description:** DDA Generic Workflow	**Region Code:** RW
☐	**Process Name:** DDAEODWF	**Version:** 1	**Process Description:** DDAEODWF	**Region Code:** RW
☐	**Process Name:** DDAPUSHCHEQUEBOOK	**Version:** 2	**Process Description:** Push cheque book	**Region Code:** RW
☐	**Process Name:** DDASTAGEUPDATEWF	**Version:** 1	**Process Description:** DDASTAGEUPDATEWF	**Region Code:** RW

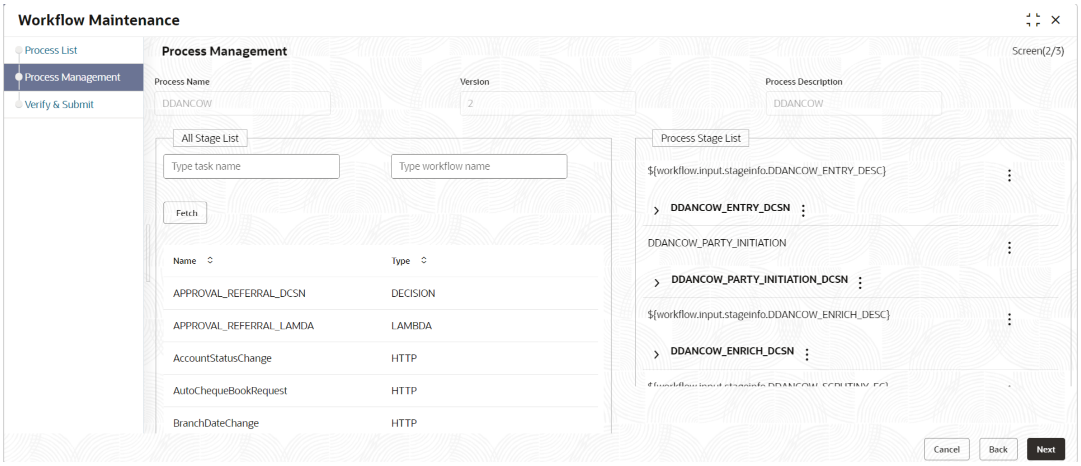
2. Select the **Process Name: blank** check box.
3. Click the **Upload DSL+** button to upload the batch process definition file.

   The File Explorer window displays.
4. Select the file `DDAEODWF.json` from the local folder.

   A new row displays with the **Process Name: DDAEODWF** in the first column.
5. Select the check-box on the new row for the **DDAEODWF** process.
6. Click **Next**.

   The **Process Management** screen displays.

Figure 1-2 Process Management



Process Name: DDANCOW
Version: 2
Process Description: DDANCOW

Name	Type
APPROVAL_REFERRAL_DCSN	DECISION
APPROVAL_REFERRAL_LAMDA	LAMBDA
AccountStatusChange	HTTP
AutoChequeBookRequest	HTTP
BranchDateChange	HTTP

${workflow.input.stageinfo.DDANCOW_ENTRY_DESC}
- DDANCOW_ENTRY_DCSN

DDANCOW_PARTY_INITIATION
- DDANCOW_PARTY_INITIATION_DCSN

${workflow.input.stageinfo.DDANCOW_ENRICH_DESC}
- DDANCOW_ENRICH_DCSN

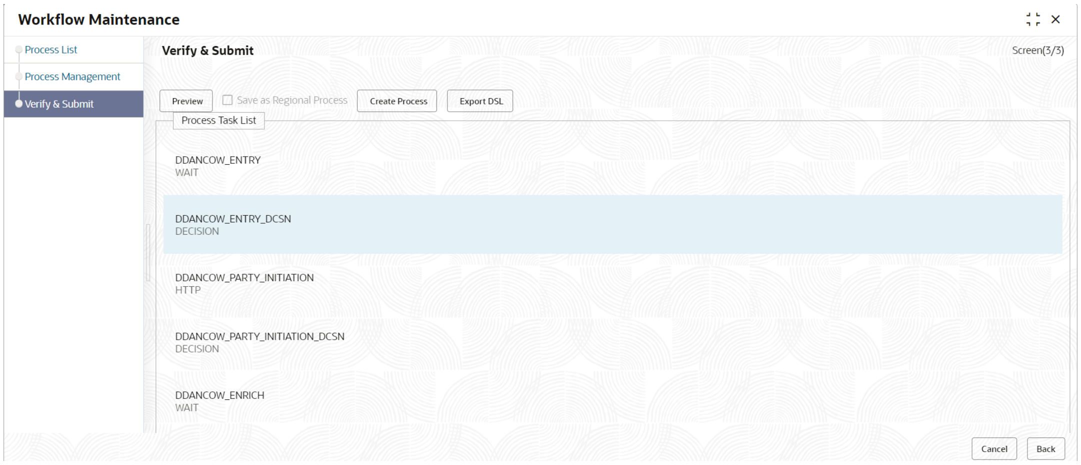
Note

You can add, modify, and delete the EOD processes from this page.

7. Click **Next** button.

   The **Verify and Submit** screen displays.

Figure 1-3 Verify and Submit



Task	Type
DDANCOW_ENTRY	WAIT
DDANCOW_ENTRY_DCSN	DECISION
DDANCOW_PARTY_INITIATION	HTTP
DDANCOW_PARTY_INITIATION_DCSN	DECISION
DDANCOW_ENRICH	WAIT

8. Click **Preview**.

   The Flow Diagram dialog displays the work flow process sequence.

9. Click **Create Process**.

   The EOD process workflow is created.

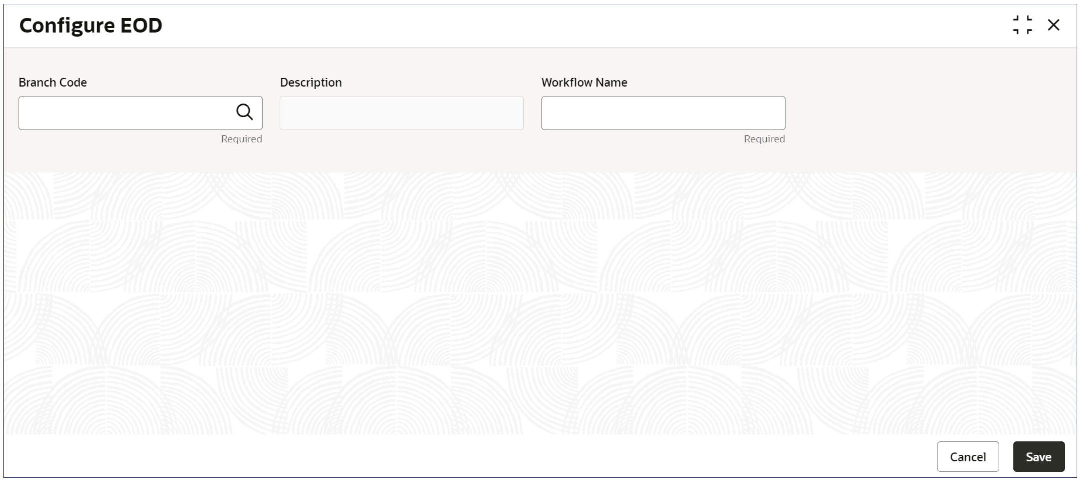
1.3 Configure EOD

This topic describes the systematic instructions to configure EOD (End of Day) operations.

1. Click **Core Maintenance**, and under **Core Maintenance**, click **Branch EOD**.
2. Under **Branch EOD**, click **Configure EOD**.

   The **Configure EOD** screen displays.

   Figure 1-4 Configure EOD

> **Note**
>
> To configure the batch for a branch, refer to the **Configure Branch EOD** section in the *Oracle Banking Common Core User Guide*.

3. Specify the details described in the following table.

   Table 1-1 Configure EOD

Field Name	Description
Branch Code	Specify the Branch Code to configure the EOD batch processes.
Description	The branch description is auto-populated.
Workflow Name	Specify a name for the EOD batch workflow. **Note:** This name must match the **workflow name** attribute value specified in the 3rd line of the batch script `DDAEODWF.json` file.

4. Click **Save**.

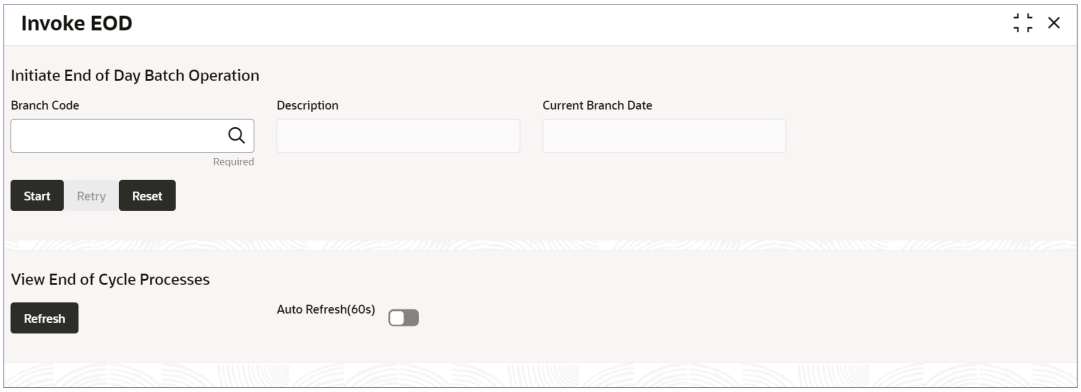
1.4 Invoke EOD

This topic describes the systematic instructions to run the EOD for a branch.

1. Click **Core Maintenance**, and under **Core Maintenance**, click **Branch EOD**, then click **Invoke EOD**.

   The **Invoke EOD** screen displays.

Figure 1-5 Invoke EOD



2. Initiate the EOD Batch Operation.
   a. Specify the fields described in the following table.

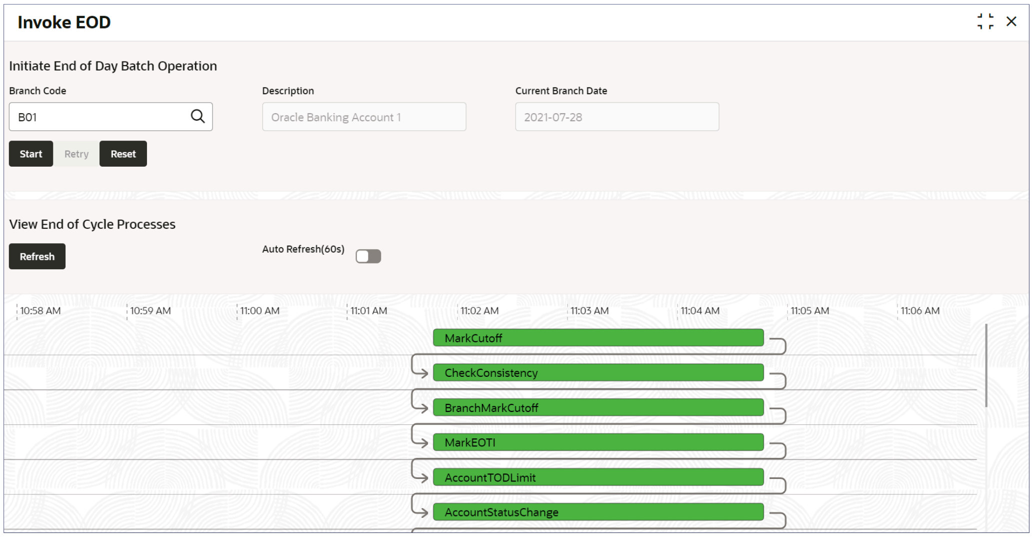
Table 1-2 Initiate End of Day Batch Operation

Field Name	Description
Branch Code	Specify the Branch Code you want to invoke the EOD batch processes.
Description	The branch description is auto-populated.
Current Branch Date	The current date is auto-populated.

   b. Click **Start**.

   The **View End of Cycle Processes** section gets populated and the end of cycle processes' status display.

Figure 1-6 Invoke EOD - View End of Cycle Processes


Branch Code: B01
Description: Oracle Banking Account 1
Current Branch Date: 2021-07-28

3. Click **Refresh** to view the latest status of the End of Cycle processes.

Note

Turn on the **Auto Refresh (60s)** option to automatically refresh the processes status every minute.

2

Oracle Banking Corporate Accounts Batch Jobs and APIs

The topic describes the Oracle Banking Corporate Accounts batch jobs and APIs.

Table 2-1   Oracle Banking Corporate Accounts Batch Jobs and APIs

Sl. No.	EOD Stage	Name	Description
1	MCUT	MarkCutoff	This API changes the branch status from Transaction Input to Cutoff in OBA.
2	MCUT	CoherencePostProcessing	This API saves the transactions present in the coherence cache.
3	MCUT	AbortExistingWithdrBalanceUpdateBatchJob	This API aborts an executing Withdrawable Balance Update job for a branch.
4	MCUT	CheckWithdrBalanceUpdateJobStatusCompleted	This API check the status as Completed for Withdrawable Balance Update job for a branch.
5	MCUT	WithdrBalanceUpdateBatchJob	This batch calculate the withdrawable balance for the account from transactions and moves the transaction to domain schema
6	MCUT	AbortExistingPostProcessingBatchJob	This API aborts an executing post-processing job for a branch.
7	MCUT	AbortExistingPostProcessingBatchJobNostro	This API aborts an executing post-processing job for a Nostro branch.
8	MCUT	CheckPostProcessingJobStatusCompleted	This API check the status as Completed for Post Processing batch job for a branch.
9	MCUT	CheckPostProcessingJobStatusCompletedNostro	This API check the status as Completed for Nostro Post Processing batch job for a branch.
10	MCUT	PostProcessingBatchJob	This batch calculates value-dated and book-dated balances of an account. It also activates the account if it is in a dormant state.
11	MCUT	PostProcessingBatchJobNostro	This batch calculates value-dated and book-dated balances of a Nostro account. It also activates the Nostro account if it is in a dormant state.
12	MCUT	AbortExistingStatementNarrativeBatchJob	This API aborts an executing Statement Narrative batch job for a branch
13	MCUT	AbortExistingStatementNarrativeBatchJobNostro	This API aborts an executing Nostro Statement Narrative batch job for a branch
14	MCUT	CheckStatementNarrativeJobStatusCompleted:	This API check the status as Completed for Statement Narrative batch job for a branch.
15	MCUT	CheckStatementNarrativeJobStatusCompletedNostro	This API check the status as Completed for NostroStatement Narrative batch job for a branch.
16	MCUT	StatementNarrativeBatchJob	This batch sends the statement narrative of a transaction to the statement domain.
17	MCUT	StatementNarrativeBatchJobNostro	This batch sends the statement narrative of a transaction to the Nostro statement domain.

Table 2-1 (Cont.) Oracle Banking Corporate Accounts Batch Jobs and APIs

Sl. No.	EOD Stage	Name	Description
18	MCUT	CheckConsistency	This API checks for completeness of all transactions received for the current branch date
19	MCUT	BranchMarkCutoff	This batch marks cutoff in Common core
20	EOTI	MarkEOTI	Marks the end of transaction input
21	EOTI	AccountTODLimitPopulate	This batch job populates the cache with the accounts eligible for a temporary overdraft or renewal of their temporary overdraft limit.
22	EOTI	AccountTODLimitExecutor	This batch job processes the accounts in the cache populated by the AccountTODLimitPopulate batch job and enables or renews the temporary overdraft limit as required by the accounts.
23	EOTI	AccountStatusChangePopulation	This batch creates a list of accounts eligible for automatic account status change. This option is provided in the Status Rule Definition of the Account class used to create the account. For more information, see Create Account Class in the Corporate Accounts User Guide.
24	EOTI	AccountStatusChangeExecution	This batch processes the accounts identified by the AccountStatusChangePopulation batch job. It computes the new account status, and updates the account status if it is different from the old account status.
25	EOTI	ICMarkCutoff	Marks the cutoff for Interest batch
26	EOTI	DDA-IC	This batch computes and liquidates Accruals and Interest for accounts on the current branch date. It posts the resulting transactions.
27	EOTI	ICCoherencePostProcessing	This API saves the DDA-IC generated transactions present in the coherence cache.
28	EOTI	ICAbortExistingWithdrBalanceUpdateBatchJob	This API aborts an executing Withdrawable Balance Update job for a branch.
29	EOTI	ICCheckWithdrBalanceUpdateJobStatusCompleted	This API check the status as Completed for Withdrawable Balance Update job for a branch
30	EOTI	ICWithdrBalanceUpdateBatchJob	This batch calculate the withdrawable balance for the account from transactions and moves the transaction to domain schema.
31	EOTI	ICAbortExistingPostProcessingBatchJob	This API aborts a currently executing branch post-processing job.
32	EOTI	ICAbortExistingPostProcessingBatchJobNostro	This API aborts a currently executing Nostro branch post-processing job.
33	EOTI	ICCheckPostProcessingJobStatusCompleted	This API check the status as Completed for Post Processing batch job for a branch.
34	EOTI	ICCheckPostProcessingJobStatusCompletedNostro	This API check the status as Completed for Nostro Post Processing batch job for a branch.
35	EOTI	ICPostProcessingBatchJob	This batch calculates the value-dated and book-dated balances of an account. It also activates the account if it is in a dormant state. This batch executes after the DDA-IC transactions are posted by the DDA-IC batch jobs.

Table 2-1 (Cont.) Oracle Banking Corporate Accounts Batch Jobs and APIs

Sl. No.	EOD Stage	Name	Description
36	EOTI	ICPostProcessingBatchJobNostro	This batch calculates the value-dated and book-dated balances of a NOSTRO account. It also activates the NOSTRO account if it is in a dormant state. This batch executes after the DDA-IC Nostro transactions are posted by the DDA-IC batch jobs.
37	EOTI	ICAbortExistingStatementNarrativeBatchJob	This API aborts an executing Statement Narrative batch job for a branch.
38	EOTI	ICAbortExistingStatementNarrativeBatchJobNostro	This API aborts an executing Nostro Statement Narrative batch job for a branch.
39	EOTI	ICCheckStatementNarrativeJobStatusCompleted	This API check the status as Completed for Statement Narrative batch job for a branch.
40	EOTI	ICCheckStatementNarrativeJobStatusCompletedNostro	This API check the status as Completed for Nostro Statement Narrative batch job for a branch.
41	EOTI	ICStatementNarrativeBatchJob	This batch sends the transaction statement narrative to the statement domain for the transactions posted in the DDA-IC jobs.
42	EOTI	ICStatementNarrativeBatchJobNostro	This batch sends the transaction statement narrative to the statement domain for the NOSTRO transactions posted in the DDA-IC jobs.
43	EOTI	ICCheckConsistency	Checks for consistency after the IC batch job complete.
44	EOTI	AccountRevaluationCleanup	This batch job cleans up the already processed revaluation data.
45	EOTI	AccountRevaluationBatch	This batch revalues the FCY (Foreign Currency) accounts based on the exchange rate defined for the current date and the revaluation setup configuration.
46	EOTI	AccountRevaluationNettedGlHandoff	This API is used to post the Netted GL Handoff for the transaction generated by the revaluation batch.
47	EOTI	AccountRevaluationNostroCleanup	This batch job cleans up the already processed revaluation data from the driver for Nostro Accounts.
48	EOTI	AccountRevaluationNostro	This batch revalues the FCY accounts based on the exchange rate defined for the current date and the revaluation setup configuration for Nostro accounts.
49	EOTI	AccountRevaluationNettedGlHandoffNostro	This API is used to post the Netted GL Handoff for the transaction generated by the revaluation batch for Nostro.
50	EOTI	RevaluationCoherencePostProcessing	This API saves the transactions present in the coherence cache for the above Account Revaluation generated transactions.
51	EOTI	RevaluationAbortExistingWithdrBalanceUpdateBatchJob	This API aborts an executing Withdrawable Balance Update job for a branch.
52	EOTI	RevaluationCheckWithdrBalanceUpdateJobStatusCompleted	This API check the status as Completed for Withdrawable Balance Update job for a branch
53	EOTI	RevaluationWithdrBalanceUpdateBatchJob	This batch calculate the withdrawable balance for the account from transactions and moves the transaction to domain schema.

Table 2-1 (Cont.) Oracle Banking Corporate Accounts Batch Jobs and APIs

Sl. No.	EOD Stage	Name	Description
54	EOTI	RevaluationAbortExistingPostProcessingBatchJob	This api aborts a currently executing post processing job for the branch.
55	EOTI	RevaluationAbortExistingPostProcessingBatchJobNostro	This API aborts aborts a currently executing NOSTRO branch post-processing job.
56	EOTI	RevaluationCheckPostProcessingJobStatusCompleted	This API check the status as Completed for Post Processing batch job for a branch
57	EOTI	RevaluationCheckPostProcessingJobStatusCompletedNostro	This API check the status as Completed for NostroPost Processing batch job for a branch.
58	EOTI	RevaluationPostProcessingBatchJob	This batch calculates the value-dated balance and book-dated balance of an account. It also activates the account if it is in a dormant state. This batch executes after the Account revaluation batch posts the account transactions.
59	EOTI	RevaluationPostProcessingBatchJobNostro	This batch calculates the value-dated balance and book-dated balance of a NOSTRO account. It also activates the NOSTRO account if it Is in a dormant state after the Account revaluation batch posted transactions for Nostro.
60	EOTI	RevaluationAbortExistingStatementNarrativeBatchJob	This API aborts an executing Statement Narrative batch job for a branch.
61	EOTI	RevaluationAbortExistingStatementNarrativeBatchJobNostro	This API aborts an executing Nostro Statement Narrative batch job for a branch.
62	EOTI	RevaluationCheckStatementNarrativeJobStatusCompleted	This API check the status as Completed for Statement Narrative batch job for a branch
63	EOTI	RevaluationCheckStatementNarrativeJobStatusCompletedNostro	This API check the status as Completed for Nostro Statement Narrative batch job for a branch
64	EOTI	RevaluationStatementNarrativeBatchJob	This batch sends the transaction statement narrative to the statement domain for the transactions posted in in Account Revaluation batch.
65	EOTI	RevaluationStatementNarrativeBatchJobNostro	This batch sends the transaction statement narrative to the statement domain for the transactions posted in the NOSTRO Account Revaluation batch.
66	EOFI	MARKEOFI	Marks the end of financial input
67	EOFI	BranchMarkEOFI	Marks the end of financial input in Common core
68	EOD	GlHandoffCleanup	This batch job removes the GL handoff data that is present from a previously executed batch.
69	EOTI	GLHandoffBatch	This batch job identifies and creates a list of Accounting transactions, Accounts with a status change, and Accounts with a balance change for current branch date that are eligible for GL handoff. It then creates the required transaction and posts it to the glhandoff.
70	EOTI	GlHandoffNostroCleanup	This batch job removes the Nostro GL handoff data that is present from a previously executed batch.

Table 2-1 (Cont.) Oracle Banking Corporate Accounts Batch Jobs and APIs

Sl. No.	EOD Stage	Name	Description
71	EOTI	GLHandoffBatchNostro	This batch job identifies and creates a list of Nostro Accounting transactions, Nostro Accounts with a status change, and Nostro Accounts with a balance change for current branch date that are eligible for GL handoff. It then creates the required transaction and posts it to the Nostro glhandoff.
72	Date Change	BranchDateChange	Changes system date to next working date in Common core.
73	Date Change	DDADateChange	Changes system date to next working date in OBA.
74	Date Change	ICDateChange	Changes system date to next working date in IC domain.
75	Date Change	ICReleaseCutoff	Releases the branch's cutoff in IC domain.
76	Date Change	BranchReleaseCutOff	Release of cutoff in Common core.
77	BOD	BranchMarkTI	Marks the transaction input for the branch.
78	BOD	DormancyBatchCleanup	This batch job removes the DormancyBatch data that is present from a previously executed batch.
79	BOD	DormancyBatch	This batch marks the account dormant if the dormancy date of the account is before the current branch date.
80	BOD	ChequebookPopulateDriverBatchJob	This batch job identifies and lists the accounts that are eligible for a chequebook auto reorder by checking the automatic reorder level of cheque leaves.\
81	BOD	ChequebookExecutorBatchJob	This batch places chequebook requests for the accounts identified by the ChequebookPopulateDriverBatchJob.
82	BOD	ReleaseUncollected	This batch releases uncollected funds for accounts which are due for release on the branch date.
83	BOD	ReleaseLegalAmountBlocks	This API releases legal amount blocks for accounts.
84	BOD	StopPaymentPopulateJob	This task identifies and creates a list of the accounts that require Stop payments.
85	BOD	StopPaymentExecutorBatchJob	This batch updates the stop_payment status of the list of accounts generated by StopPaymentPopulateJob.
86	BOD	StatementPopulateDriverBatchJob	This batch identifies and creates a list of the accounts which are eligible for today's statement generation.
87	BOD	StatementExecutorBatchJob	This batch generates the statements for list of accounts generated by StatementPopulateDriverBatchJob.
88	BOD	StatementPopulateDriverBatchJobNostro	This batch identifies and creates a list of the Nostro accounts that are eligible for today's statement generation.
89	BOD	StatementExecutorBatchJobNostro	This batch generates the statements for list of accounts generated by StatementPopulateDriverBatchJobNostro.

3

Batch Description

The topic provides information on the various **Oracle Banking Corporate Accounts Cloud Service (OBCACS)** batch jobs.

1. Account Status Change

The batch updates the status of an account automatically if the flag **status_change_automatic** in the account is enabled. The OBA application maintains various statuses of accounts. The rule engine maintains the rules to determine an account status and status rules are then attached to the account class for every stage movement. The batch picks such accounts where the **status_change_automatic** flag is enabled and evaluates the status of an account using the rules maintained in the account class. The system evaluates and updates the new status of an account.

The status change of an account refers to modifications that change the account type, account ownership, or updates to account information based on financial or non-financial transactions.

2. Account Revaluation

Revaluation is a calculated upward adjustment to a country's official exchange rate relative to a chosen baseline. The revaluation batch process revalues the balances of foreign currency customer accounts and restates the local currency (LCY) balance. The system uses the batch results to revalue an account's balance and posts the revaluation profit or loss into a predefined account. The system then hands over the revaluation profit or loss to the GL system.

If the **Reval Split Required** option is enabled, then a trading split in revaluation is required for the GL. The break-up of the revaluation profit or loss for the GL can be::

- **Trading Profit / Loss** – Profit or loss due to revaluation of Foreign Currency (FCY) entries posted into the FCY account during the day.
- **Revaluation P&L** – Profit or loss due to revaluation of opening balances (balances without current day’s turnover).

Based on the configurations, the system books profits and losses to the Profit GL and Loss GL. If the **Reval Split Required** option is enabled, the system books the profits and losses to the Trading Profit GL and Trading Loss GL respectively.

3. Account Balance Handoff to GL System

The EOD batch process performs the following:

- Hands over the accounting entries posted to a GL account to the GL system.
- Hands over the account balance to the reporting GL. The batch process retrieves the reporting Credit and Debit GL based on the account's new status and account balance.

Note

The system maintains the reporting Credit GL Line and Debit GL Line for every possible account status in the account class or the account. It posts a positive account balance to the Credit GL and a negative account balance to the Debit GL. An Inter-system bridge GL maintained as a source code preference facilitates balance posting. The system posts the offset entries for each scenario to the Inter-system Bridge GL.

The following conditions are handled by the batch process.

- No change in the balance sign and the account has net credit turnover.
- No change in the balance sign and the account has net debit turnover.
- No change in the account balance, as there are no transactions for the day.
- No change in the account balance, since the net turnover (sum of debits and credits) is zero.
- Net credit turnover in the account changing the account's balance sign from negative to positive.
- Net debit turnover in the account changing the account's balance sign from positive to negative.

4. Dormancy Batch

The **Dormancy** batch marks all accounts without financial or non-financial activity for a specified number of days as dormant. The Account class maintains the non-financial activities and other parameters that determine the dormancy of an account. The system sets the `last_cr_activity_date` and `last_dr_activity_date`. The batch process determines the dormancy date of an account from the following:

```
[Greater of (last_cr_activity_date, last_dr_activity_Date)] +
 Dormancy_days (Obtained from the account class) + 1
```

The Dormancy batch picks all accounts which are not dormant, and whose dormancy date is lesser than the branch date; and marks it dormant.

5. Automatic Reorder of Cheque Books

The EOD batch process executes a batch function to determine the automatic reordering of cheques. It initiates automatic reorder of a chequebook for an account when the following conditions are satisfied:

- The **Auto-Reorder of Cheque Book** option is enabled in the Account.
- The number of unused cheque leaves for the account is less than or equal to the reorder level maintained in the Account.
- The number of cheque leaves in the new Chequebook is equal to the **Reorder Number of Leaves** maintained in the account.

The system determines the number of leaves in the chequebook from the field **Reorder Number of Leaves** maintained for the account. The numbering of the cheque leaves in the new Cheque Book depends on the **Cheque Number Unique for Branch** option in the **Bank Parameters** maintenance. If this option is enabled, the numbering begins from the **Last Number + 1** of the Chequebook last delivered to any account in the branch. If you do not select this option, the number will start from the **Last Number +1** of the previous Chequebook delivered to the same account.

6. Legal Block Expiry Batch

The Legal Block Expiry Batch releases the legal amount blocks placed on an account on the expiry date. The batch performs the following:

- Picks all the accounts with legal blocks expiring earlier or on the branch date.
- Derives the value of the valid legal block to retain.
- Expires the Legal Block and updates the account balance.

7. Stop Payment Batch

This batch job updates an account's stop payment status to **Yes** or **No**.

- Getting Expired Stop Payments - Closes all stop payments for the branch date and if there are no active stop payments for the account, it updates the account's stop payment status to *Yes*.
- Activating Stop Payments - Updates the stop payment flag in the account to *Yes* when there are active stop payments for the account on the branch date.

TOD Batch

The TOD Batch runs at the Beginning of the Day (BOD) and updates the **Temporary Overdraft (TOD)** limit for accounts. For each account reaching the TOD expiry date, it performs the following:

- If the **TOD_renew** option is not enabled, it resets the TOD limit to zero.
- If the **TOD_renew** option is enabled, it derives and sets the new TOD limit including the TOD limit renewal amount.

8. Statement View

The statement batch runs as BOD (Beginning of Day) and takes the PWD (Previous working date) as the input parameter. For example, when EOD runs on 31-Jan-2023, the statement batch is executed as the BOD of 01-Feb-2023, fetching those accounts for which the statement is due on 31-Jan-2023.

The EOD statement batch generates Statements, Advices, SWIFT, and CAMT messages as specified in the statement preferences. The statement preferences specify the frequency of statement generation and the statement types of primary, secondary, and tertiary statements.

The system captures multiple addresses for an account. Address Type classifies these addresses as Head Office Address (HOA), Branch Office Address (BOA), Communication Address (COA), and others. For every Address Type, different Media type address can be maintained, like Email, Mail, SWIFT, and FAX.

Statements are linked and delivered to different media addresses associated with the different address types. Statements are delivered in PDF format for all media types except ISO and SWIFT.

OBCACS supports predefined Statement and Advice format for the detailed and summary versions of each statement type. Since the batch runs at the Beginning Of the Day (BOD), it takes the previous business day as input and generates all statements due.

The **Report Format** configuration maintains the format of the advices and statements. And the **Report Linkage** configuration links the statements and advices to the report formats. You can customize the statement or advice format to include predefined advice tags. To know more, see Customize Statements and Advices.

A

Customize Statements and Advices

Use the predefined XML elements described in this topic to customize the information in the account statements and advices.

This topic contains the following sections:

- Predefined XML elements
- Sample XML file

Predefined XML elements

Table A-1 Predefined XML elements description table

XML Element	Description
statementAdviceXML	root element
reportHeaderXML	child element
customerName	Customer Name
customerNumber	Customer Number
reportGenerationDate	Date when report is generated
addressLine1	Address Line 1
addressLine2	Address Line 2
addressLine3	Address Line 3
addressLine4	Address Line 4
pageHeaderXML	child element
fromDate	From Date
toDate	To Date
detailXML	child element
accountsXML	child element
accountXML	child element (Complex Type) - could be repeated for consolidated statements
accountName	Account Name
accountNo	Account Number
branchCode	Branch Code
branchName	Branch Name
currency	Account Currency
currentBalance	Current Balance
openingBalance	Opening Balance
closingBalance	Closing Balance
apyeAdviceRequestXML	child element (Specific to Retail Accounts)
apyeAvgBal	Average balance in the retail account
apye	child element (Specific to Retail Accounts)
liqdAmt	Available balance in the retail account
transactionItems	XML child element

Table A-1 (Cont.) Predefined XML elements description table

XML Element	Description
item	XML child element (Complex Type) - could be repeated for multiple transaction
accClass	Account Class of the Account
accountCcy	Currency Of the Account
accountCcyAmount	Transaction Amount in Account Currency
accountEntryType	Indicates whether Entry is associated with Account(A) or GL(G)
accountingRefNo	Uniquely generated reference number to identify each transaction
accountingRefNoToBeReversed	Accounting reference number which is being reversed
accountNumber	Account Number
accountType	Indicates the Type of Account for which the Transaction is being posted
addlText	Indicates short explanation for each transaction to be shown in statements
amtInRealAccCcy	Transaction Amount in Account Currency
amtTag	Indicates Amount tag Associated with the Transaction
authStatus	Authorization Status of the Transaction
autoRelease	Flag to determine the release of the uncollected credit transaction on the uncollected fund release date
availabilityInfo	Indicates the time interval after which funds will be available.
availableDays	Indicates the number of days after which funds will be available
balance	Running Balance
balancedEntry	Indicates the sum of debits and sum of credits to be balanced if this flag is On
bankRefNo	Bank Reference Number
blockRefNo	Amount block reference number associated with the transaction
blockUtilizedAmt	Block Amount utilised
branchLcyAmtTransaction	Amount in Transaction Branch Currency
courtesyPaySource	Courtesy Pay Source associated with the transaction
customerNo	Customer Number
deferredResponse	Specify yes to defer the response.
deltaAmt	The difference in the amount,
deposit	Transaction Amount in Account Currency
drCr	Debit Credit Indicator
eaRefNo	External Accounting Reference Number
event	Transaction Event
exchRate	Exchange Rate
extAccountingRefNo	External Accounting Reference Number
externalChecker	User who has authorized the transaction
externalMaker	User who has Initiated the transaction
fileName	File Name associated with the Amount Block
forcePost	Flag to Indicate that Amount block has to be created irrespective of any error
instrumentCode	Instrument Code associated with the Amount block
limitRequired	Limit to be utilized

Table A-1 (Cont.) Predefined XML elements description table

XML Element	Description
module	Indicates the Module from which the transaction is being posted
multiCcyAccNo	Indicates if it is a Multi-currency account number
operation	Operation name
originalSource	Source of transaction origination
pendingAmt	Pending Amount
product	Product Code associated with the Transaction
productProcessor	Product Processor which is sending the Transaction
realAccCcy	Account Currency
realAccClass	Account Class of the Account
realAccountBrn	Branch associated with the Account
realAccountNo	Account Number
referralAllowed	Referral allowed or not
regdApplicable	Indicates whether the transaction is applicable for Federal Reserve Registered Regulation
relatedAccount	Account associated with the transaction
relatedCustomer	Customer associated with the transaction
relatedReferrence	Related Reference Number associated with the Transaction
reval	Flag to Indicate Revaluation is to be done for the Transaction
revalCode	Revaluation Code to be associated with the Transaction being posted
revalLossGl	Revaluation Loss GL to be associated with the transaction being posted
revalProfitGl	Revaluation Profit GL to be associated with the transaction being posted
revalRate	Revaluation Rate to be associated with the transaction being posted
revalRateCode	Revaluation Rate Code to be associated with the transaction being posted
revalRequired	Indicates whether revaluation is required for the transaction being posted
revalTxnCode	Revaluation Transaction Code to be associated with the transaction being posted
reversal	Flag to Indicate whether the transaction is a reversal Transaction
rtlAllowed	Indicates if RTL is allowed on the transaction or not
rtlRefNo	RTL reference number
showInStmt	Indicator whether to show in statement or not
sourceCode	Source system from where the transaction is initiated
sourceRefNo	Source reference number
statementNarrative	Indicates short explanation for each transaction to be shown in statements
status	Status
stmtDate	Date on which the transaction is recorded in the statement generated for the account
suppressAccOverridesUptoSeverityLevel	Severity level upto which can be tolerated for the account validation override suppression in a transaction
suppressBalanceOverridesUptoSeverityLevel	Severity level upto which can be tolerated for the balance validation override suppression in a transaction
tag61SupportInfo	Tag 61 Info to be associated with the transaction
transactionAmount	Transaction amount

Table A-1 (Cont.) Predefined XML elements description table

XML Element	Description
transactionCcy	Currency in which the transaction is done
transactionDate	Transaction Posting Date
transactionDescription	Indicates short explanation for each transaction to be shown in statements
transactionStatus	Transaction status of the transaction posted
txnBranch	Transaction Branch from which Transaction is initiated
txnDescription	Transaction Description
txnInitDate	Transaction Initialization Date
txnLcy	Transaction amount in LCY
txnRefNo	Transaction Reference Number
txnStatus	Transaction status of the transaction posted
userRefNo	User Reference Number
valueDate	Value Date of the Transaction
withdrawal	Transaction Amount in Account Currency

Sample XML file

The following sample XML file shows the use of the predefined XML tags described in the Predefined XML elements table.

```
<statementAdvice>
    <pageHeader>
       <fromDate>2018-04-07</fromDate>
       <toDate>2018-04-07</toDate>
    </pageHeader>
    <reportHeader>
       <customerName>Barclays</customerName>
       <customerNumber>001</customerNumber>
       <reportGenerationDate>2018-04-07 08:04:00</reportGenerationDate>
       <addressLine1>36/12 Umesh Mukherjee Road</addressLine1>
       <addressLine2>Belgharia</addressLine2>
       <addressLine3>Kolkata</addressLine3>
       <addressLine4>INDIA 123456</addressLine4>
    </reportHeader>
    <detail>
       <accounts>
          <account>
             <accountName>VA123</accountName>
             <accountNo>123</accountNo>
             <branchCode>000</branchCode>
             <branchName>BR000</branchName>
             <currency>GBP</currency>
             <currentBalance>500</currentBalance>
             <openingBalance>3000</openingBalance>
             <closingBalance>500</closingBalance>
             <transactionItems>
                <item>
                   <transactionDate>2018-04-08 08:04:00</transactionDate>
                   <valueDate>2018-04-07 08:04:00</valueDate>
                   <transactionDescription>Some test value</
transactionDescription>
                   <withdrawal>2500</withdrawal>
                   <deposit></deposit>
                   <balance>500</balance>
                   <accClass></accClass>
                   <accountCcy></accountCcy>
                   <accountCcyAmount></accountCcyAmount>
                   <accountEntryType></accountEntryType>
                   <accountingRefNo></accountingRefNo>
                   <accountingRefNoToBeReversed></accountingRefNoToBeReversed>
                   <accountNumber></accountNumber>
                   <accountType></accountType>
                   <addlText></addlText>
                   <amtInRealAccCcy></amtInRealAccCcy>
                   <amtTag></amtTag>
                   <authStatus></authStatus>
                   <autoRelease></autoRelease>
                   <availabilityInfo></availabilityInfo>
                   <availableDays></availableDays>
                   <balancedEntry></balancedEntry>
                   <bankRefNo></bankRefNo>
                   <blockRefNo></blockRefNo>
                   <blockUtilizedAmt></blockUtilizedAmt>
                   <branchLcyAmt></branchLcyAmt>
                   <courtesyPaySource></courtesyPaySource>
                   <customerNo></customerNo>
                   <deferredResponse></deferredResponse>
                   <deltaAmt></deltaAmt>
                   <drCr></drCr>
                   <eaRefNo></eaRefNo>
                   <event></event>
                   <exchRate></exchRate>
                   <extAccountingRefNo></extAccountingRefNo>
                   <externalChecker></externalChecker>
                   <externalMaker></externalMaker>
                   <fileName></fileName>
                   <forcePost></forcePost>
                   <instrumentCode></instrumentCode>
                   <limitRequired></limitRequired>
                   <module></module>
                   <multiCcyAccNo></multiCcyAccNo>
                   <operation></operation>
                   <originalSource></originalSource>
                   <pendingAmt></pendingAmt>
                   <product></product>
                   <productProcessor></productProcessor>
                   <realAccCcy></realAccCcy>
                   <realAccClass></realAccClass>
                   <realAccountBrn></realAccountBrn>
                   <realAccountNo></realAccountNo>
                   <referralAllowed></referralAllowed>
                   <regdApplicable></regdApplicable>
                   <relatedAccount></relatedAccount>
                   <relatedCustomer></relatedCustomer>
                   <relatedReferrence></relatedReferrence>
                   <reval></reval>
                   <revalCode></revalCode>
                   <revalLossGl></revalLossGl>
                   <revalProfitGl></revalProfitGl>
                   <revalRate></revalRate>
                   <revalRateCode></revalRateCode>
                   <revalRequired></revalRequired>
                   <revalTxnCode></revalTxnCode>
                   <reversal></reversal>
                   <rtlAllowed></rtlAllowed>
                   <rtlRefNo></rtlRefNo>
                   <showInStmt></showInStmt>
                   <sourceCode></sourceCode>
                   <sourceRefNo></sourceRefNo>
                   <statementNarrative></statementNarrative>
                   <status></status>
                   <stmtDate></stmtDate>
                   <suppressAccOverridesUptoSeverityLevel></
suppressAccOverridesUptoSeverityLevel>
                   <suppressBalanceOverridesUptoSeverityLevel></
suppressBalanceOverridesUptoSeverityLevel>
                   <tag61SupportInfo></tag61SupportInfo>
                   <transactionAmount></transactionAmount>
                   <transactionCcy></transactionCcy>
                   <transactionStatus></transactionStatus>
                   <txnBranch></txnBranch>
                   <txnDescription></txnDescription>
                   <txnInitDate></txnInitDate>
                   <txnLcy></txnLcy>
                   <txnRefNo></txnRefNo>
                   <txnStatus></txnStatus>
                   <userRefNo></userRefNo>
                </item>

             </transactionItems>
          </account>
       </accounts>
    </detail>
</statementAdvice>
```

Index